

AMENDMENTS TO THE SPECIFICATION

Replace the two paragraphs from page 9, line 20 to page 10, line 20 with the following paragraphs:

Process business artifact 103 is a workflow process definition that describes the mechanism by which a business operation is to be carried out. More specifically, process business artifact 103 describes a number of activities that are to be used to solve a particular business problem and a number of rules that govern the operation flow between those activities. In an exemplary embodiment, a workflow process definition may be constructed using a software tool called Rational Rose (<http://www.rational.com/products/rose/index.jsp>) for visually modeling the activities and rules that form a workflow process definition.

Once a workflow process definition associated with process business artifact 103 is formed, the workflow process definition is received by a workflow framework business artifact service 113 that transforms the workflow process definition into software code that forms the code skeleton of the desired business application. Referring now to FIG. 7, there is shown an activity diagram 701 for checking the validity of an address that may be part of a workflow process definition, according to an exemplary embodiment. In addition to a start node 703 and a stop node 705, activity diagram 701 includes a ValidateAddress automatic activity 707, an AddressValid automatic activity 709 and an AddressInvalid automatic activity 711. Also included in activity diagram 701 is a

FixInvalidAddress manual activity 713 and a CreateConfirmation subprocess 715. In an exemplary embodiment, workflow framework business artifact service 113 receives activity diagram 701 and generates the code example contained in Exhibit A of the Appendix. Workflow framework 113 may forward engineer a workflow process definition into any type of software code suitable for developing a business application including, by way of non-limiting example, Java code. The process of transforming a workflow process definition into software code may be performed using Together ControlCenter from TogetherSoft (<http://www.togethersoft.com/products/controlcenter/index.jsp;jsessionid=mtfwcentl.wwww6>).

Replace the paragraph on page 14, lines 3-7 with the following paragraph:

Platform independent services 117 interfaces with any of a plurality of platform adapters 125 so that the resulting business application may execute on and interface with a corresponding operating environment. Platform adapters 125 may include, by way of non-limiting example, adapters for interfacing the business application to a Windows® NT platform, an IBM® S390 platform and a Solaris platform.

Replace the paragraph on page 15, lines 9-17 with the following paragraph:

Reporting application pattern 300 is used for building applications in which information is delivered to any output device such as, by way of non-limiting example, a web-browser, a printer or an application program (for example, an Excel® spreadsheet). In an exemplary embodiment, the reporting application pattern includes tools such as, by way of non-limiting example, style sheets that may be used for conforming reports to any desired look and feel. In order to build business applications that conform to reporting application pattern 300, only user-interface business artifact 107 and data business artifact 111 are used. Exhibit G of the Appendix includes an example of a user-interface business artifact and data artifact associated with a reporting application for responding to transaction inquiries.

Replace the paragraph from page 19, line 19 to page 20, line 19 with the following paragraph:

A number of embodiments of the present invention have been described. Nevertheless, it will be understood that various modifications may be made without departing from the spirit and scope of the invention. Based on the above description, it will be obvious to one of ordinary skill to implement the system and methods of the present invention in one or more computer programs that are executable on a

programmable system including at least one programmable processor coupled to receive data and instructions from, and to transmit data and instructions to, a data storage system, at least one input device, and at least one output device. Each computer program may be implemented in a high-level procedural or object-oriented programming language, or in assembly or machine language if desired; and in any case, the language may be a compiled or interpreted language. Suitable processors include, by way of example, both general and special purpose microprocessors. Furthermore, alternate embodiments of the invention that implement the system in hardware, firmware or a combination of both hardware and software, as well as distributing modules and/or data in a different fashion will be apparent to those skilled in the art and are also within the scope of the invention. In addition, it will be obvious to one of ordinary skill to use a conventional database management system such as, by way of non-limiting example, Sybase, Oracle and DB2, as a platform for implementing the present invention. Also, network access devices can comprise a personal computer executing an operating system such as Microsoft WindowsTM, UnixTM, or Apple Mac OSTM, as well as software applications, such as a JAVA® program or a web browser. Network access devices 203-205 can also be a terminal device, a palm-type computer, mobile WEB access device or other device that can adhere to a point-to-point or network communication protocol such as the Internet protocol. Computers and network access devices can include a processor, RAM and/or ROM memory, a display capability, an input device and hard disk or other relatively

Appln. No. 10/037,938

PATENT
Attorney Docket No.: 74577-013

permanent storage. Accordingly, other embodiments are within the scope of the following claims.